

**REMARKS**

The following remarks form a full and complete response to the Office Action dated July 9, 2009. Claims 11-31 remain pending in the present application (claims 1-10 having been previously cancelled without prejudice or disclaimer). Applicants submit them for reconsideration.

The Examiner rejected claims 11-31 under 35 U.S.C. § 102(e) as anticipated by U.S. Patent No. 6,742,175 to Brassard et al. (“Brassard”). Applicants traverse the rejection on the basis claims 11-31 recite subject matter not disclosed by Brassard.

***Claims 11-24 and 29***

In response to Applicants’ previous argument that “Brassard merely discloses a single code generator able to target different languages one at a time and not several code generators, as claim 11 requires,” the Examiner alleges that Brassard discloses several code generators as supposedly seen in Fig. 3 of Brassard. *See* Office Action at 2. Applicants disagree.

The Examiner first points out that in Fig. 3, element 50 is used to generate the model declaration 51. However, the term “*generation*” in the state of the art is used to designate an automated process of creation of any content, as opposed to a work done manually through the intervention of a human user. Fig. 3 vividly shows the need of human intervention in 50, as it clearly relates the action of 50 to the existence of user input through keyboard 53 and mouse 54. *See* Brassard at FIG. 3. Applicants, therefore, submit that element 50 cannot be considered part of the generator, but is rather model declaration software operated by a human user to create the model. No generation takes place at this point and, therefore, Applicants submit that the Examiner’s assertion that several code generators are “used to translate the part of the description that the generators provide” is inexact. Instead, at this point, the generator is not yet in use (i.e.,

visual modeling tool 50 is not part of the generator itself) and no "part" is provided here, because, after component 50 is activated, the human user must manually operate 49 (code generator UI) to make it be fed with the model declaration 51 that a human user created using manually 50. Then again, no distribution, no part of a description and no generator exist here before 52 is activated through 51. Applicants also note that 51 is only a UI, meaning that the model 50 is directly sent to the generation engine 52 along with generation instructions also defined by the human user. Thus, neither model tool 50 nor model declarations 51 constitute code generators.

The Examiner also suggests that model declaration engine 46, generation instruction engine 47, and recursion binder 48 are several code generators. They are not.

First of all, the Applicants wish to stress the fact that, in the present application, the several code generators each receive a part of the model declaration, this being achieved by a model distribution realized using modifiable distribution rules. No such distribution is realized in Brassard for the simple reason that there are not several generators to distribute a part of a description to. On the contrary, Brassard's generator engine is unique and composed of several parts that are not by themselves generators. *See* Brassard at col. 10, line 53 to col. 11, line 49.

With regard to recursion binder 48, Brassard notes Recursion binder 48 is explicitly described by Brassard as a part of generator engine 52 and not as a code generator by itself. *See* Brassard at col. 11, lines 5-34. More precisely, 48 is said to be a recursion binder replacing parameter strings by value strings within a part of the code already generated by 52. Recursion binder 48, therefore, does not generate any code; it merely modifies it afterwards, thus showing that the Examiner's assertion that 48 is one of several code generators is erroneous.

Furthermore, generation instructions engine 47, contrary to the Examiner's assertions, is merely a code parser that only analyzes the generation instructions that parameterize the

generator engine 52 and only passes them on to 52. *See* Brassard at col. 10, lines 53-62. As such, generation instructions engine 47 does not generate any code and only provides 52 with code segments that are to be assembled by 52. And this is the unique generation engine 52 which is ultimately in charge of actually generating the source code. Thus, neither 47 nor 48 is a code generator. Again, code generation is done by 52.

To further bolster this point, Applicants note that Brassard states that “the generator engine 52 is therefore composed of a model declaration engine 46, a generation instructions engine 47 and a recursion binder 48”. *See* Brassard at col. 11, lines 41-43. As such, generator engine 52 is the only code generator described in Brassard and 46, 47 and 48 are only components thereof. Consequently, none of the these components of code generators are code generators themselves, nor are they distributed any part of a description. Distribution means that each receiving element of the distributed parts has its own part to process that is not sent to the other receiving elements. This concept contains a notion of parallelism. In Brassard, the aforementioned components of 52 sequentially receive each code string. Thus, this is a serialized process, and not a parallel one, that involves generated code segments and not parts of a description. In brief, in the above-cited portions of Brassard, Brassard gives examples of use of these engines where they are called sequentially to build iteratively the same part of the generated source code, thus clearly showing the absence of any distribution between these parts of the generator engine.

Thus, Brassard fails to disclose distributing a description between several code generators, as required by claim 11. For this reason alone, the rejection of claim 11 is improper and should be withdrawn.

In addition to the reasons stated above, claim 11 is patentable over Brassard for the separate and independent reason that Brassard fails to disclose distributing the description between several code generators according to modifiable distribution rules, as claim 1 requires.

In response to the argument that Brassard fails to disclose the concept of modifiable distribution rules, the Examiner suggests that Brassard's FIG. 8 (and the accompanying discussion) discloses this concept. *See* Office Action at 3. Applicants disagree.

The Examiner seems to compare modifiable distribution rules and modification of generation instructions, assuming that these terms are related to the same concept. These two concepts are, however, different from each other.

Distribution rules are rules meant to describe how distribution is to take place. In the context of the present application, this distribution is the action of distributing parts of a description of a computer application to several code generators. By contrast, the generation instructions that are modified by the user in Brassard are a part of the elements that the generation engine 52 is provided with to generate a source code. Indeed, the source code generator disclosed by Brassard would have no need of distribution rules because, as noted above, Brassard's code generation involves only one code generator, which would hardly require any distribution rules. In addition, Brassard's generation instructions are technology-dependent definitions that describe the software components that Brassard generator assembles. *See* Brassard at col. 11, line 50 to col. 12, line 57. A person of ordinary skill in the art would understand that this kind of generation instructions do not describe any kind of distribution at all; rather, they describe interfaces to components that are assembled by the code generator.

Applicants also note that there is a fundamental difference between the invention embodied in the present application and what is disclosed in Brassard. According to the present invention, the modifiable rules are a means of generating from its description the same

application on different target architectures. Brassard, by contrast, describes a process of regenerating previously generated source code that has been modified by a developer since its generation, preserving these changes in the process of regeneration while applying those changes in the component binding that are required by the newly modified generation instructions

What Brassard describes has nothing to do with distribution of parts of a software application to several code generators as previously demonstrated, and therefore, the modification of generation instructions only changes the way in which the unique generator engine 52 performs its generation on a single language at a time and cannot be considered to be a "*modification of a distribution rule*" because, apart from the obvious difference in the terms used to name this modifiable element, no distribution process takes place in Brassard. Thus, Claim 11 patentable over Brassard for the separate and independent reason that Brassard fails to disclose modifiable distribution rules.

Since Brassard fails to disclose (1) generating computer code by distributing a description between several code generators (2) according to modifiable distribution rules, as required by claim 11, Applicants respectfully request the withdrawal of the rejection of claim 11. Claims 12-24 depend, either directly or indirectly, from claim 11 and are patentable for at least the same reasons stated above with respect to claim 11 as well as for the additional features they recite. Applicants, therefore, respectfully request the withdrawal of the rejection of claims 12-24 as well.

Claim 29 is patentable over Brassard because, similarly to claim 11, it recites (1) generating computer code by distributing a description between several code generators (2) according to modifiable distribution rules. As stated above with respect to claim 11, Brassard entirely fails to disclose these features of claim 29. Applicants, therefore, respectfully request the withdrawal of the rejection of claim 29.

***Claims 25-28 and 30-31***

In response to the previous Office Action, Applicants argued that Brassard failed to disclose “services that cannot be defined by the language,” as required by claims 25, 27, 30, and 31. The Examiner argues that Brassard discloses a software description language in which the other classes cannot have access to any one of technical or functional service except through a first class. *See* Office Action at 4. The Examiner further points out that Brassard discloses a software description language with classes.

The Applicants fully agree with the Examiner’s unremarkable assertion that Brassard discloses a software description language with classes (indeed, the present application discusses classes in prior art definitions mentioned in the application). That Brassard discloses a software description language with classes is, however, of no moment to the patentability of the present application because the difference between the prior art and the invention as claimed is based on something else: that this language does not give access to certain services, whether technical or functional, except through first classes. This was explained on pages 10 and 11 of our response to the previous Office Action.

The Examiner also suggests that Brassard discloses a software description language with first class giving exclusive access to certain services. Applicants disagree. Within the context of the present application, a person of ordinary skill in the art would clearly understand “first classes” to involve the concept exclusive access to certain services. *See, e.g., Specification<sup>1</sup> at ¶ 46.* Brassard does not disclose this concept.

---

<sup>1</sup> Unless otherwise noted, citations to the Specification refer to U.S. Patent Application Publication No. 2005/0177817, published on August 11, 2005.

In response to Applicants' previous arguments that Brassard fails to disclose "services that cannot be defined by the language", the Examiner refers us to portions of Brassard relating to a Universal Modeling Language (UML) and to a "class diagram." These portions of Brassard, however, in no way relate to the relevant features of the claimed invention. Indeed, the Examiner's reference to these portions of Brassard as showing the relevant features of claims 25 and 27 seems to reveal a slight misunderstanding of the claimed invention; the Examiner appears to allege that the UML is the same as the software description language and that it is both prohibited from defining technical and functional services and that it only has access to these services through the "core services" classes illustrated in FIG. 11 and described in Example 1, beginning on col. 24, line 19 of Brassard. *See* Office Action at 4.

As shown in Brassard's Fig. 1, however, Brassard discloses a library of pre-defined templates 32, which does not exclude the possibility of delivering the same service through a redefinition described fully with the description language. Again, this comment applies to Fig. 11, which shows core services for a banking application, where the illustrated classes only define classes that will be generated without preventing the user of this language from defining the exact same service directly with the example description language. For a person skilled in the art, the only fact that the modeling language used is UML implies straightforwardly that any kind of service, whether technical or functional, can be described with it.

Thus, Brassard fails to disclose a software description in which service cannot be defined by the language, as required by claims 25, 27, and 30-31. To summarize, all the examples included in the Examiner's arguments show samples of description language, or samples of use of such a language, but none of these examples evokes either explicitly or implicitly the restriction enforced by this language to access certain services. Thus Brassard fails to disclose each and every feature of claims 25, 27, and 30-31. Applicants, therefore, respectfully request

the withdrawal of the rejection of claims 25, 27, and 30-31. Claims 26 and 28 depend from claims 25 and 27, respectively, and are patentable over Brassard for at least the same reasons stated above with respect to claims 25 and 27 as well as for the additional features they recite. Applicants, therefore, respectfully request the withdrawal of the rejection of claims 26 and 28 as well.

### CONCLUSION

In view of the above, all objections and rejections have been sufficiently addressed. Applicants submit that the application is now in condition for allowance and request that the Office allow claims 11-31 and pass this application to issue.

If any additional payment is required, please charge the cost thereof to deposit account no. 02-2135.

If for any reason the Examiner determines that the application is not now in condition for allowance, it is respectfully requested that the Examiner contact, by telephone, the Applicants' undersigned attorney at the indicated telephone number to arrange for an interview to expedite the disposition of this application.

Respectfully submitted,

Date: November 9, 2009

By: /John H. Curry/

John H. Curry  
Registration No. 65,067  
ROTHWELL, FIGG, ERNST & MANBECK  
1425 K Street, N.W., Suite 800  
Washington, D.C. 20005  
Telephone: (202)783-6040  
Facsimile: (202) 783-6031